IN THE CLAIMS

Please amend the claims as follows.

For the Examiner's convenience, a list of all claims is included below.

1. (Currently Amended) A method for execution by a microprocessor in response to receiving a single instruction, the method comprising:

receiving a first vector of numbers from a first entry in a register file and a second vector of numbers from a second entry in the register file;

selecting a first plurality of numbers of the first vector from the first entry and a second plurality of numbers of the second vector from the second entry according to a configuration specified by the instruction; and

generating simultaneously a third plurality of numbers, each of which is an absolute difference between a number in the first plurality of numbers and a number in the second plurality of numbers;

wherein the third plurality of numbers are saved in a third entry in the register file;

wherein the above operations are performed in response to the microprocessor receiving the single instruction, wherein the single instruction indicates the first entry for the first plurality of numbers, the second entry for the second plurality of numbers, and the third entry for the third plurality of numbers in the register file.

2. (Original) A method as in claim 1 wherein an absolute difference between a first number and a second number is computed using a method comprising:

producing a first intermediate number by subtracting the second number from the first number;

producing a second intermediate number by subtracting the first number from the second number; and

selecting a positive number from the first intermediate number and the second intermediate number as the absolute difference between the first number and the second number;

wherein the microprocessor is a media processor disposed on an integrated circuit with a memory controller.

3. (Original) A method as in claim 2 wherein the first intermediate number and the second intermediate number are produced in parallel; and wherein the third plurality of numbers are generated substantially simultaneously.

4. (Previously Presented) A method as in claim 2 further comprising:

testing if an overflow occurs in producing the first intermediate number and the second intermediate number;

saturating the absolute difference between the first number and the second number if an overflow occurs.

5. (Previously Presented) A method as in claim 1 wherein the first plurality of numbers are received from a first entry in a register file.

6. (Previously Presented) A method as in claim 5 wherein the single instruction specifies a way to partition a string of bits in the first entry into the first plurality of numbers.

7. (Previously Presented) A method as in claim 5 wherein the single instruction specifies an index of the first entry in the register file.

8. (Canceled)

9. (Previously Presented) A method as in claim 1 wherein the single instruction specifies an index of the entry in the register file.

10. (Original) A method as in claim 1 wherein a type of each of the first and second pluralities of numbers is one of:

a) unsigned integer;

b) signed integer; and

c) floating point number.

11. (Original) A method as in claim 1 wherein a size of each of the first and second pluralities of numbers is one of:

a) 8 bits;

b) 16 bits; and

c) 32 bits.

12. (Currently Amended) A machine readable media containing an executable computer program instruction which when executed by a digital processing system causes said system to perform a method comprising:

receiving a first vector of numbers from a first entry in a register file and a second vector of numbers from a second entry in the register file;

selecting a first plurality of numbers of the first vector from the first entry and a second plurality of numbers of the second vector from the second entry according to a configuration specified by the instruction; and

generating simultaneously a third plurality of numbers, each of which is an absolute difference between a number in the first plurality of numbers and a number in the second plurality of numbers;

wherein the third plurality of numbers are saved in third entry in the register file;

wherein the above operations are performed in response to a microprocessor of the digital processing system receiving the instruction, wherein the instruction indicates the first entry for the first plurality of numbers, the second entry for the second plurality of numbers, and the third entry for the third plurality of numbers in the register file.

13. (Original) A media as in claim 12 wherein an absolute difference between a first number and a second number is computed using a method comprising:

producing a first intermediate number by subtracting the second number from the first number;

producing a second intermediate number by subtracting the first number from the second number;

selecting a positive number from the first intermediate number and the second intermediate number as the absolute difference between the first number and the second number.

14. (Original) A media as in claim 13 wherein the first intermediate number and the second intermediate number are produced in parallel.

15. (Previously Presented) A media as in claim 12 wherein the method further comprises:

testing if an overflow occurs in producing the first intermediate number and the second intermediate number;

saturating the absolute difference between the first number and the second number if an overflow occurs.

16. (Previously Presented) A media as in claim 12 wherein the first plurality of numbers are received from a first entry in a register file.

17. (Previously Presented) A media as in claim 16 wherein the single instruction

specifies a way to partition a string of bits in the first entry into the first plurality of numbers.

18. (Previously Presented) A media as in claim 16 wherein the single instruction specifies an index of the first entry in the register file.

19. (Canceled)

20. (Previously Presented) A media as in claim 12 wherein the single instruction specifies an index of the entry in the register file.

21. (Original) A media as in claim 12 wherein a type of each of the first and second pluralities of numbers is one of:

a) unsigned integer;

b) signed integer; and

c) floating point number.

22. (Original) A media as in claim 12 wherein a size of each of the first and second pluralities of numbers is one of:

a) 8 bits;

b) 16 bits; and

c) 32 bits.

23. (Currently Amended) An execution unit in a microprocessor, the execution unit comprising:

a first circuit configured to receive a first vector of numbers from a first entry in a register file;

a second circuit configured to receive a second vector of numbers from a second entry in the register file;

a selector circuit configured to select a first plurality of numbers of the first vector from the first entry and a second plurality of numbers of the second vector from the second entry according to a configuration specified by the instruction; and

a third circuit coupled to the first circuit and the second circuit, the third circuit, in response to the microprocessor receiving the single instruction, generating simultaneously a third plurality of numbers, each of which is an absolute difference between a number in the first plurality of numbers and a number in the second plurality of numbers, wherein the instruction indicates the first entry for the first plurality of numbers, the second entry for the second plurality of numbers, and the third entry for the third plurality of numbers in the register file,

wherein the microprocessor is a media processor disposed on an integrated circuit with a memory controller.

24. (Original) An execution unit as in claim 23 wherein the third circuit comprises a plurality of units, each of the plurality of units comprising:

a fourth circuit configured to generate a first intermediate number by subtracting a number in the second plurality of numbers from a number in the first plurality of numbers;

a fifth circuit configured to generate a second intermediate number by subtracting a number in the first plurality of numbers from a number in the second plurality of numbers; and

a sixth circuit coupled to the fourth circuit and the fifth circuit, the sixth circuit selecting a positive number from the first intermediate number and the second intermediate number as an absolute difference between the first number and the second number.

25. (Currently Amended) A processing system, comprising:

an execution unit comprising

a first circuit configured to receive a first vector of numbers from a first entry in a register file;

a second circuit configured to receive a second vector of numbers from a second entry in the register file;

a selector circuit configured to select a first plurality of numbers of the first vector from the first entry and a second plurality of numbers of the second vector from the second entry according to a configuration specified by the instruction; and

a third circuit coupled to the first circuit and the second circuit, the third circuit, in response to the microprocessor receiving the single instruction, generating

simultaneously a third plurality of numbers, each of which is an absolute difference between a number in the first plurality of numbers and a number in the second plurality of numbers, wherein the single instruction indicates the first entry for the first plurality of numbers, the second entry for the second plurality of numbers, and the third entry for the third plurality of numbers in the register file, wherein the microprocessor is a media processor disposed on an integrated circuit with a memory controller.

26. (Currently Amended) An execution unit in a microprocessor, the execution unit comprising:

means for receiving a first vector of numbers from a first entry in a register file and a second vector of numbers from a second entry in the register file;

means for selecting a first plurality of numbers of the first vector from the first entry and a second plurality of numbers of the second vector from the second entry; and

means for generating simultaneously a third plurality of numbers, each of which is an absolute difference between a number in the first plurality of numbers and a number in the second plurality of numbers;

wherein the third plurality of numbers are saved in a third entry in the register file;

wherein the above means perform operations in response to the microprocessor receiving a single instruction, wherein the single instruction indicates the first entry for the first plurality of numbers, the second entry for the second plurality of numbers, and the third entry for the third plurality of numbers in the register file, and wherein the selecting is performed according to a configuration specified by the instruction.

27. (Original) An execution unit as in claim 26 wherein an absolute difference between a first number and a second number is computed using a unit comprising:

means for producing a first intermediate number by subtracting the second number from the first number;

means for producing a second intermediate number by subtracting the first number from the second number;

means for selecting a positive number from the first intermediate number and the second intermediate number as the absolute difference between the first number and the second number.

28. (Original) An execution unit as in claim 27 wherein the first intermediate number

and the second intermediate number are produced in parallel.

29. (Previously Presented) An execution unit as in claim 27 further comprising:

means for testing if an overflow occurs in producing the first intermediate number and the second intermediate number;

means for saturating the absolute difference between the first number and the second number if an overflow occurs.

30. (Previously Presented) An execution unit as in claim 26 wherein the first plurality of numbers are received from a first entry in a register file.

31. (Previously Presented) An execution unit as in claim 30 wherein the single instruction specifies a way to partition a string of bits in the first entry into the first plurality of numbers.

32. (Previously Presented) An execution unit as in claim 30 wherein the single instruction specifies an index of the first entry in the register file.

33. (Canceled)

34. (Previously Presented) An execution unit as in claim 30 wherein the single instruction specifies an index of the first entry in the register file.

35. (Original) An execution unit as in claim 26 wherein a type of each of the first and second pluralities of numbers is one of:

a) unsigned integer;

b) signed integer; and

c) floating point number.

36. (Original) An execution unit as in claim 26 wherein a size of each of the first and second pluralities of numbers is one of:

a) 8 bits;

b) 16 bits; and

c) 32 bits.

37. (Previously Presented) A method as in claim 1, wherein a type of each of the first and second pluralities of numbers is floating point number.

38. (Previously Presented) A media as in claim 12, wherein the microprocessor is a media processor disposed with a memory controller on an integrated circuit.

39. (Previously Presented) An execution unit as in claim 26 further comprising:

means for testing if an overflow occurs.

40. (Previously Presented) An execution unit as in claim 23, wherein the memory controller is usable to access memory not disposed on the integrated circuit.

41. (Previously Presented) An execution unit as in claim 40, wherein the memory controller is usable by a host central processing unit (CPU) not disposed on the integrated circuit to access the memory.